

Laux in view of Vange, U.S. patent application publication 2002/0002603 (hereinafter “Vange”). Applicants are traversing the rejections.

Traversal

The rejection of claims 6 and 19 for lack of enabling disclosure

The language “the request includes a specifier referring to an object that is not present in the particular database system” of claims 6 and 19 is supported in overview at page 12, line 6-page 13, line 14:

**Modifying a distributed database system with a materialized view site so that the materialized view site can function as a queryable cache:
FIG. 2**

A materialized view site is like a queryable cache in that it responds to SQL requests and moves a subset of the database objects contained in a master site closer to the consumer of the data. It is unlike a queryable cache in that it is not transparent; when a data object is missing in the materialized view site but present in the master site, a SQL request that refers to the data object simply fails, rather than being automatically redirected to the master site. FIG. 2 shows how a distributed database system with a materialized view site can be modified to make the materialized view site function as a queryable cache.

Database system 201 in FIG. 2 includes a redirecting database system 203 which is a modified materialized view site and a redirection target database system 229 which is a master site for redirecting database system 203. Redirecting database system 203 has the global name DBS_1 and redirection target database system 229 has the global name DBS_2. At the highest level, database system 201 operates as follows:

1. Database system 203 receives an SQL request 204;
2. Database system 203 attempts to execute the SQL request; as part of executing the SQL request, database system 203 parses the request. As part of parsing the request, database system 203 determines whether the request can be executed in database system 203 or must be redirected to database system 229. In the former case, database system 203 executes the request and returns the data 206 resulting from the execution to the source of the SQL request;
3. If the request must be redirected, a *miss* has occurred and database system 203 redirects part or all of the SQL request to database system 229, as shown by arrow 231;
4. Database system 229 executes the redirected portion of the SQL request and returns the results to database system 203 (arrow 233).

which integrates it with any part of the results obtained in database system 203 and returns the complete results 206 to the source of SQL request 204.

There are many reasons why a miss may occur. Perhaps the most common is that *an object referred to in the request is not present in database system 203*; another is that the request updates an object and the object is read only in database system 203 but updateable in database system 229; a third is that the request uses information about an object that may be different in database system 203 and in database system 229; a fourth is that the user making the request has privileges in database system 203 that prohibit the request and privileges in database system 229 that allow it. (emphasis added)

If the foregoing is not sufficient to establish that the limitation “the request includes a specifier referring to an object that is not present in the particular database system” is fully supported by the Specification as filed, Applicants respectfully refer Examiner to the detailed description of a preferred embodiment of redirection set forth at page 15, line 31-page 18 line 6.

The rejection of claims 5-7, 10-12, 15, 16, 19, 21, and 24 under 35 U.S.C. 102(e) as anticipated by Laux

5 *What Applicants are claiming*

In order for a reference to anticipate a claim, the reference must disclose within its four corners all of the limitations of the claim. Applicants’ independent claim 5 as amended in the *Submission* that accompanied Applicants’ RCE is exemplary for the limitations of Applicants’ independent claims:

- 1 5. (previously presented) A method employed in a distributed database system
- 2 that includes a plurality of database systems for responding to a request
- 3 *received in a particular database system of the plurality*, each database
- 4 system of the plurality including a query engine and a database and
- 5 the method comprising the steps *performed in the particular database system’s*
- 6 *query engine of:*
- 7 determining *when the request is parsed whether an execution of the*
- 8 *request is preferably done at least in part in another database system of the*
- 9 *plurality; and*
- 10 *if that is the case, redirecting that part of the execution to the other*
- 11 *database system.* (emphasis added)

The distributed database system is shown in FIG. 2; distributed database system 201's "plurality of database systems" is embodied in DBS 203 and DBS 229; each of these DBS's has query engine 205 and a database 220. DBS 203 receives SQL request 204 and thus embodies the claim's "particular database system". The method of the claim is "performed in the particular database system's query engine" 205. Details of query engine 205 are shown in FIG. 3, where the parser may be seen at 301. An embodiment of the behavior described in the two method steps is seen at 305, 321, 323, 307, 319, and 231 and described at page 16, line 15-page 17, line 24. An important aspect of what is being claimed is that when a miss occurs, the execution of the part of the query that caused the miss is redirected *by the particular database system* to the *other database system*, i.e., the redirection is from one database system to another database system, as may be clearly seen from arrow 234 in FIG. 2. The results of the redirected execution are returned to the particular database system as shown by arrow 233. Of course, as pointed out at page 13, lines 18-24, the other database system may again redirect part or all of the query that it has received from the particular database system.

The disclosure of Laux

About all that Laux has to do with the system of Applicants' claim 5 is that in both Laux' system and Applicants' system, execution of a query may involve more than one database system. Of course, when more than one database system is involved, the execution of the query must deal with differences in the database systems. Laux is concerned with techniques for dealing with differences in the database systems' *drivers*; Applicants' techniques, by contrast, deal with differences in the *contents* of the database systems: as set forth in Applicants' claim 5, when the "query engine" of the "particular database system" that received the request "determin[es] whether the execution of the request is preferably done at least in part in another database system of the plurality", "if that is the case, that part of the execution" is "redirect[ed] to the other database system".

What Laux discloses is not a system in which a query engine in one database system redirects a query on miss to another database system, but rather a system which dispatches a query to a number of different data sources more or less simultaneously and

constructs a single result from the results it receives from the different data sources. As described at [0031] of Laux,

Thus, merging driver 125 performs a kind of demultiplexing operation for any access operation received by merging driver 125. Any corresponding instruction or sequence of instructions is distributed by merging driver 125 to the different data sources through their corresponding drivers and their common API, as required.

As shown clearly in Laux' FIG. 2, there is no redirection between database systems in Laux. The data sources in FIG. 2 are the data sources of FIG. 1; as shown in FIG. 2, merging driver 125 receives a query from user interface 190 and distributes queries based on the query received from user interface 190 to each of the data sources 140-170; there is no interaction at all between the data sources 140-170, and since that is so, the data sources 140-170 neither "determin[e] ... whether an execution of the request is preferably done at least in part in another database system of the plurality" nor "if that is the case, redirect[] that part of the execution to the other database system".

There is no indication whatever in Laux that the form of the query which is received by a particular one of his data sources is determined by the *contents* of the data source. The problem being solved by Laux is simply that the data sources have different drivers, and the form of the query is determined at least in part by the driver. See in this regard [0038]:

in distribute query operation 320, merging driver 125 converts the query received from user interface API 205 to a plurality of queries that includes a query that can be passed to each of drivers 120 to 123 using driver API 220i. Merging driver 125 sends the query to each driver API 220i and then transfers to obtain and order results operation 330.

It appears from the above that the queries received by the different data sources are logically equivalent. Each data source responds to the query it receives by returning whatever it has of the information specified in the query. There is simply nothing in Laux that has anything to do with redirect on miss. Confirmation of that fact is provided by the fact that searches of Laux on "redirect" and on "miss" return no result.

It is further clear from FIG. 1 and FIG. 2 that merge driver 125 is not a component of any of the data sources 140-170. Moreover, the steps of the methods of FIG. 3 and FIG. 4 are performed by the “merging driver” (see [0037]-[0048]) and are thus not performed in any of data sources 140-170. Applicants’ claim 5 by contrast requires that the method steps be performed “in the particular database system’s query engine”.

As is apparent from the foregoing, Laux does not disclose most of the limitations of claim 5 and consequently cannot anticipate the claim. Examiner will immediately see that the arguments just made with regard to claim 5 apply equally to claim 15. Because neither of the independent claims is anticipated by Laux, none of the dependent claims can be anticipated by Laux.

Independent patentability of the dependent claims

As one would expect from the differences between Laux and Applicants’ independent claims, Laux also does not disclose any of the limitations which are added in claims 6-9 and 16-19. In claims 6 and 19, the added limitation is to the specific redirection that occurs “when the object required for execution of the request is not present in the particular database system”. As pointed out above, there is no notion of either misses or redirection in Laux. In claims 7 and 16, the form of the request is modified “in the query engine” when it is determined that the request will be redirected. Again, Laux’ query modification is not done in the query engines of his data sources and does not occur on redirection. Examiner himself admits that the additional limitations of claims 8 and 17 are not disclosed in the references.

The rejection of claims 9, 14, 18, and 23 under 35 U.S.C. 103 as obvious over the combination of Laux and Vange

The rejections of claims 9 and 18 are completely dependent on the 102 rejections of claims 5 and 15; moreover, there is no disclosure in either Laux or Vange of rewriting of procedure calls on redirection. Consequently, the combined references do not disclose the added limitation of these claims and the rejection is without basis.

The rejection of the Beauregard claims 12-14 and 20-25.

Beauregard claims 12-14 and 20-24 are of course patentable because the claims they are dependent from are patentable.

Conclusion

Applicants have traversed all of Examiner's rejections: they have demonstrated that claims 6 and 19 are fully supported by the Specification as originally filed, have demonstrated that Laux does not anticipate claims 5-7, 10-12, 15, 16, 19-21, and 24 and that the combination of Laux and Vange does not disclose the added limitations of claims 9 and 18. Applicants have thereby satisfied the requirements of 37 C.F.R. 1.111(b). Applicants consequently respectfully request that Examiner continue with the examination as provided by 37 C.F.R. 1.111(a) and allow the claims as amended. No fees are believed to be required for this response; should any be, please charge them to deposit account number 501315.

Respectfully submitted,
/Gordon E. Nelson/
Attorney of record,
Gordon E. Nelson
57 Central St., P.O. Box 782
Rowley, MA, 01969,
Registration number 30,093
Voice: (978) 948-7632
Fax: (978) 945-5550
email: genelson@comcast.net
Date: 8/24/10